Poisson Regression

Timothy Hanson

Department of Statistics, University of South Carolina

Stat 506: Introduction to Experimental Design

Poisson regression

- Regular regression data $\{(x_{i1}, \ldots, x_{ip}, y_i)\}_{i=1}^n$, but now y_i is a positive integer, often a count: new cancer cases in a year, number of monkeys killed, etc.
- Predictors can be factors (categorical) or continuous, just like "regular" regression and logistic regression.
- For Poisson data, $\text{var}(y_i) = E(y_i)$; variability increases with predicted values. In regular regression, this manifests itself in the "megaphone shape" for r_i versus predicted $\hat{y}_i$.
- If you see this shape, consider whether the data could be Poisson.
- Any count, or positive integer could potentially be approximately Poisson. In fact, binomial data where n_i is really large, is approximately Poisson.

Poisson regression model

Let $y_i \sim \text{Pois}(\mu_i)$.

The **log function** relates μ_i to $\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$:

$$y_i \sim \text{Pois}(\mu_i), \quad \log \mu_i = \beta_0 + x_{i1}\beta_1 + \cdots + x_{i,p}\beta_p,$$

yielding what is commonly called the **Poisson regression** model.

Interpretation

The model can be rewritten:

$$y_i \sim \text{Pois}(\mu_i), \quad \mu_i = e^{\beta_0+\beta_1 x_{i1}+\cdots+\beta_p x_{ip}},$$

or simply $y_i \sim \text{Pois}\left(e^{\beta_0+\beta_1 x_{i1}+\cdots+\beta_p x_{ip}}\right)$.

Say we have $p = 3$ predictors. The mean satisfies

$$\mu(x_1, x_2, x_3) = e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\beta_3 x_3}.$$

Then increasing x_2 to $x_2 + 1$ gives

$$\mu(x_1, x_2 + 1, x_3) = e^{\beta_0+\beta_1 x_1+\beta_2(x_2+1)+\beta_3 x_3} = \mu(x_1, x_2, x_3)e^{\beta_2}.$$

In general, increasing x_j by one, but holding the other predictors the constant, increases the mean by a factor of e^{β_j}.

Butterflies

Extension researchers set up garden plots with different suites of plants, with each suite identified as a level of the variable Garden below. In September, they counted the number of monarch butterflies in each garden plot.

```
Input = ("
Garden   Monarchs
A        0
A        4
A        2
A        2
A        0
A        6
A        0
A        0
B        5
B        9
B        7
B        5
B        7
B        5
B        9
B        5
C       10
C       14
C       12
C       12
C       10
C       16
C       10
C       10
")
```

Butterflies

Usual approach assumes normal data within each garden:

```
d=read.table(textConnection(Input),header=TRUE)
d

boxplot(Monarchs~Garden,data=d)
f=lm(Monarchs~Garden,data=d)
anova(f)
par(mfrow=c(2,2))
plot(f)
shapiro.test(rstudent(f)) # too many zeroes!

library(ggplot2)
ggplot(d,aes(Monarchs,fill=Garden))+geom_histogram(position="dodge")
```

Butterflies

Poisson regression just as easy!

```
library(car)
library(lsmeans)
f=glm(Monarchs~Garden,family="poisson",data=d)
summary(f)

exp(1.3122) # 3.7 times more Monarchs in B vs. A
exp(1.9042) # 6.7 times more Monarchs in C vs. A
exp(1.9042-1.3122) # 1.8 times more Monarchs in C vs. B

Anova(f,type=3)
pairs(lsmeans(f,"Garden"))
```

Helicopter service data

An operations analyst in a sheriff's department studied how frequently their emergency helicopter was used during a particular year by shift 2am–8am, 8am-2pm, 2pm-8pm, 8pm-2am. A random sample of 20 counts were obtained (in time order).

```
d=read.table("http://people.stat.sc.edu/hansont/stat506/helicopter.txt",
 header=F)
counts=d[,1]
shift=factor(d[,2])

f=lm(counts~shift)
par(mfrow=c(2,2))
plot(f)
```

Uh oh! Can try a Box-Cox transformation (need to add one to each count first though), or else just analyze the data as Poisson! Let's keep going...

Offsets

- Sometimes counts are collected over different amounts of time, space...
- For example, we may have numbers of new cancer cases per *month* from some counties, and per *year* from others.
- If time periods are the same from for all data, then μ_i is the mean count per time period.
- Otherwise we specify μ_i as a rate per unit time period and have data in the form $\{(\mathbf{x}_i, y_i, t_i)\}_{i=1}^n$ where t_i is the amount of time that the y_i accumulates over. $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})$.
- Model: $y_i \sim \text{Pois}(t_i \mu_i)$.
- Have

$$y_i \sim \text{Pois}\left(e^{\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \log(t_i)}\right).$$

$\log(t_i)$ is called an *offset*.

Ache monkey hunting

Data on the number of capuchin monkeys killed by $n = 47$ Ache hunters over several hunting trips were recorded; there were 363 total records.

The hunting process involves splitting into groups, chasing monkeys through the trees, and shooting arrows straight up.

Let y_i be the total number of monkeys killed by hunter i of age a_i ($i = 1, \ldots, 47$) over several hunting trips lasting different amounts of days; total number of days is t_i. Let μ_i be the hunter i's kill rate (per day).

$$y_i \sim \text{Pois}(\mu_i t_i),$$

where

$$\log \mu_i = \beta_0 + \beta_1 a_i + \beta_2 a_i^2.$$

A quadratic effect is included to accommodate a "leveling off" effect or possible decline in ability with age. Of interest is when hunting ability is greatest; hunting prowess contributes to a man's status within the group.

Aiming for...

...dinner!

R code

```
age=c(67,66,63,60,61,59,58,57,56,56,55,54,51,50,48,49,47,42,39,40,
 40,39,37,35,35,33,33,32,32,31,30,30,28,27,25,22,22,21,20,18,17,
 17,17,56,62,59,20)
kills=c(0,0,29,2,0,2,3,0,0,3,27,0,7,0,3,0,6,1,0,7,4,1,2,2,0,0,19,9,
 0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0)
days=c(3,89,106,4,28,73,7,13,4,104,126,63,88,7,3,56,70,18,4,83,15,
 19,29,48,35,10,75,63,16,13,20,26,4,13,10,16,33,7,33,8,3,13,3,62,4,
 4,11)

f=glm(kills~age+I(age^2),offset=log(days),family="poisson")
summary(f)

rawrate=kills/days
fit2=loess(rawrate~age) # nonparametric estimate of kill rate
age.grid=seq(17,67,1)
pred2=predict(fit2,age.grid)
plot(age.grid,pred2,type="l",xlab="Age",ylab="Kill Rate")
points(age,rawrate)
fitted=exp(cbind(rep(1,length(age.grid)),age.grid,age.grid^2)%*%f$coef)
lines(age.grid,fitted,lty=2)
```

The fitted *monkey kill rate* is $\mu(a) = \exp(-5.4842 + 0.1246a - 0.0012a^2)$. At what age, typically, is monkey hunting ability maximized?

Comment on blocking

Recall that we discussed *blocking* on individuals to reduce variability. The Ache hunters actually took part in many hunting trips, i.e. there are repeated measures on each hunter. We can instead consider hunting trip j from hunter i of length L_{ij} days, and posit a mixed model

$$y_{ij} \sim \text{Pois}(\lambda_{ij} L_{ij}), \quad \log(\lambda_{ij}) = \beta_0 + \beta_1 a_i + \beta_2 a_i^2 + u_i,$$

where

$$u_1, \ldots, u_{47} \overset{iid}{\sim} N(0, \sigma^2)$$

are random *hunter ability* effects.

This model, fit in `glmer` in the `lme4` package, reduces variability by appropriately blocking the repeated measures on hunter.

Random hunter effects in R

Needed to use $a_i - 45$ instead of a_i; R complained. Sometimes have to "center" variables around some value (usually the mean) if going to include them as quadratic functions.

```
library(lme4) # has glmer function in it

d=read.table("http://people.stat.sc.edu/hansont/stat506/ache.txt",header=F)
d # look at original data set
id=d[,2]; age=d[,3]; kills=d[,4]; days=d[,5]

f=glmer(kills~I(age-45)+I((age-45)^2)+(1|id),offset=log(days),
 family="poisson")
summary(f)
```

Note p-value for quadratic effect now significant! Blocking gives you more power to zoom in on fixed effects.

Not all counts need to be modeled as Poisson!

Recall the salable flowers example from our ANCOVA notes...

```
variety= factor(c(1,1,1,1,1,1,2,2,2,2,2,2,1,1,1,1,1,1,2,2,2,2,2,2))
moisture=factor(c(1,1,1,1,1,1,1,1,1,1,1,1,2,2,2,2,2,2,2,2,2,2,2,2))
yield= c(98,60,77,80,95,64,55,60,75,65,87,78,71,80,86,82,46,55,76,68,43,47,62,70)
plotsize=c(15, 4, 7, 9,14, 5, 4, 5, 8, 7,13,11,10,12,14,13, 2, 3,11,10, 2, 3, 7, 9)
d=data.frame(yield,plotsize,variety,moisture)
plot(yield~plotsize,col=rep(1:4,each=6),main="yield by plotsize & variety:moisture",pch=19)
legend(3,90,legend=c("1:1","2:1","1:2","2:2"),col=1:4,pch=19)
f1=lm(yield~plotsize+variety*moisture,data=d)
Anova(f,type=3)
f2=lm(yield~plotsize+variety+moisture,data=d)
pairs(lsmeans(f2,"variety"))
pairs(lsmeans(f2,"moisture"))

# plotsize is the area of the plot the flowers were counted in...
f3=glm(yield~variety+moisture,offset=log(plotsize),
    family="poisson")
```

Normal model actually fits great; Poisson regression loses power to detect treatment differences.